Remarks

The amendment to claims 1 and 9 incorporates the limitations of claims 2 and 10 respectively and add the limitation regarding the arithmetic operation which is supported by the disclosure at page 12, line 17. The amendment to claims 6 and 14 recast those claims in independent form and add the limitation regarding the arithmetic operation which is supported by the disclosure at page 12, line 17.. Applicants enclose a revised Figures 7 and 8 where the label "Prior Art" has been added. Applicants submit that the amendment does not add any new matter to the disclosure.

Applicants submit that the drawings are now in compliance with MPEP 808.02(g).

The invention centers on circuits and methods which are especially useful for boosting the speed of Montgomery multiplication (which uses a large-digit number such as a 1024-bit number) without the need for extraordinary circuits such as a three port memory. The invention involves the discovery that memory access is a bottleneck which can be overcome by simultaneous reading from at least two memories in a way which is coordinated with the overall pipeline process employed by the arithmetic unit, a multiplier adder where a multiplication addition is performed in a single cycle, in performing the Montgomery multiplication.

Claims 1-16 are rejected under 35 U.S.C. 102(b) as being anticipated by Murakami et al. (U.S. 5,155,852).

Murakami et al. (U.S. 5,155,852) discloses a digital information coding system which evenly distributes valid input data to digital signal processors

operating in parallel. In Murakami et al., independent operations are executed by multiple PAUs in parallel, and the two arithmetic units, multiplier and the arithmetic/logic operation unit shown in the Fig. 21, process independent data streams as described in Table 1, Table 2, Table 3, and Table 4. Murakami et al.'s PAU can calculate only two operations in one clock cycle, one is the "operator" which calculates A+B, and the other is the "multiplier" which calculates AxB. The result of "operator" should be sent back to the input port of "multiplication" or written back to the memory block indicated as 424, 425, and 426. The result of "multiplication" should also be sent back to the input port of "operator" or written back to the memory block. In Murakami et al., "operator" and "multiplier" are separated by the register 439 and 441. To calculate x1+x2.x3+x4, Murakami et al. needs more than 4 clock cycles as follows.

1st cycle : x2 x x3	operated by multiplier 440, and the result is stored in register 441)
2nd cycle : x1 + (x2 x x3)	operated by operator 438, and the result is stored in register 439
3rd cycle: (x1 + (x2 x x3)) x 1	operated by multiplier 440, and the result is stored in register 441 to feedback to the operator
4th cycle: ((x1 + (x2 x x3)) x 1) + x4	operated by operator 438, and the result is stored in register 439

As described above, two different arithmetic units, 438 and 440, are separated, and thus the multiplication-addition operation as presently claimed in claims 1, 6, 9 and 14 cannot be executed by Murakami et al. during the same operating cycle. Thus, Murakami et al. does not disclose the claimed the use of a multiplier adder arithmetic unit, nor an arithmetic unit capable of functioning as required by the present claims.

For the above reasons, applicants submit that the claims are allowable over the prior art of record and that the application is now in condition for allowance. Such allowance is earnestly and respectfully solicited.

Respectfully submitted,
Kohji Takano et al.

By

Steven Capella, Attorney
Reg. No. 33,086
Telephone: 845-894-3669

Ser No. 10/023,141 Amendment

JP9 – 2000 – 0304 – JP1
7/8

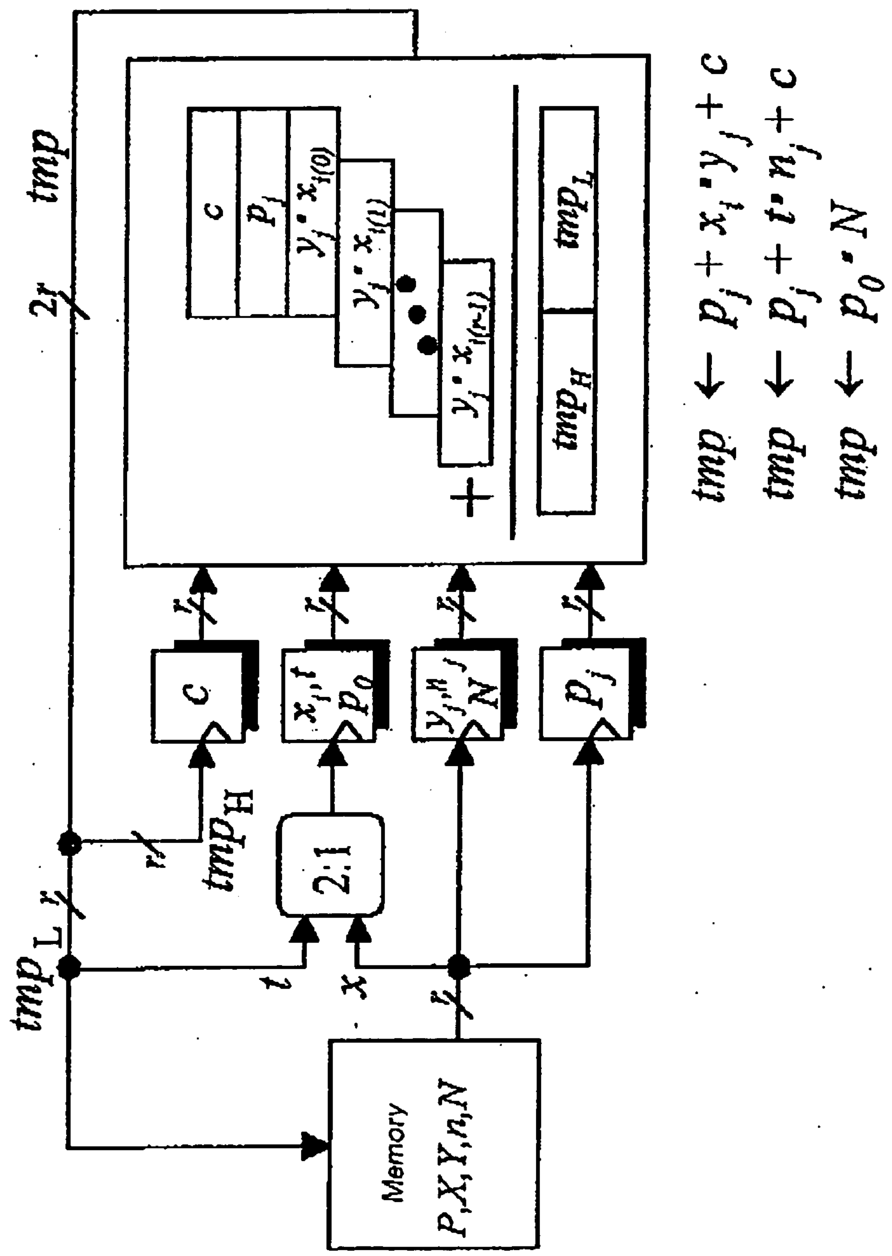

Fig. 7

Prior Art

Amendment

JP9 – 2000 – 0304 – JP1
8/8

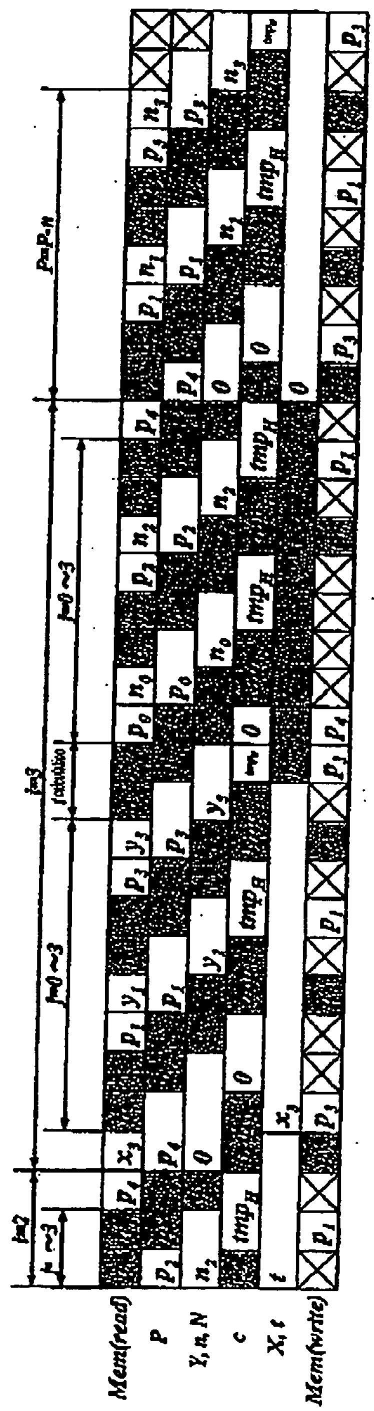

Fig. 8

Prior Art